



# **Comparative Analysis of Java and Python in Machine Learning: Investigate and Compare the Suitability and Performance of Java and Python for Machine Learning Tasks**

*Atishay Jain*

**Dr. Akhilesh Das Gupta Institute of Technology and Management (GGSIPU)**

Email Id: [Jain.atishay9@gmail.com](mailto:Jain.atishay9@gmail.com)

DOI: <https://doi.org/10.55248/gengpi.4.1223.123305>

## **ABSTRACT –**

Coding and programming are all around us and it will keep getting further wide in the hereafter. One carriage way to make a career in the ground of IT is by mastering to program. Choosing a beginner- inseparable programming language is a truly important and critical choice for a beginner in the path of getting a programmer. However, also one could truly easily lose interest in the field of programming, If a beginner is to choose a complex language when just starting in this field. There are multitudinous programming languages that it becomes truly blurring for a beginner to name which tongue to start. This document includes a comparison of the two most popular, top- ranked, and in- demand programming languages Python and Java. A compact overview of Python and Java involving its features, operations, advantages, and drawbacks is presented in this work. The comparison of the two languages is predicated on syntax and features comparison. The paper also includes the performance of a simple quick kind algorithm and a game called Tic- Tac- Toe in both the mentioned languages. The comparison result will also be concentrated on the rate of lines of law (LOC), train capacity, and speed. We are to complete which programming will be greatly better for a beginner.

**Key Words: Comparison, Python, Java, Programming Language, Beginner.**

## **1. INTRODUCTION**

The idea that programming is an inborn particularity is a myth. When tutoring how to read, we generally take shorter books including simple, abecedarian words and not a classic novel. Also, we just need to apply the same in programming also. Working easy and simple problems helps in erecting the confidence to break complicated problems. Computational Thinking is essential especially for a person associated with Computer Science.

The argument to determine which programming language to be chosen by a neophyte has been an on- going contestation Programming Language is the language used to communicate between machines and mortal beings. Computers don't understand mortal language so we need a language that machine generally understands. It gives instructions to the computer what to do next so as to perform a task or break a problem. The first programming language officially proposed was Plankalkül, developed by Konrad Zuse for his Z1 computer between 1943 and 1945 although not enforced for the time being. The first performing high- position programming language is known as the Short law proposed by John Mauchly's in 1949 and was written in the early 1950s. Fortran( Formula Translation) was the first commercially available language developed by a platoon at IBM in the time of 1954. Programming and Coding are frequently used as exchangeable words, but we need to understand that knowing how to decode doesn't mean you can program. Coding is the capability to write law with the knowledge of the syntax and structure of a language while Programming is the capability to transcribe an idea to break a problem and it comprises rendering, assaying and enforcing algorithms, understanding data structures, and working problems.

There's a large compass for a Computer Science degree holder and they can strive in any field. Currently, there are numerous programming languages available and each one of them is better than the other in their own aspects. To take exclusively one programming language is leathery. There are numerous different factors to be considered when choosing a language. Trip and misinformation of characteristics of a programming language like third party support, ease of understanding, speed or functioned. Can have a huge impact when opting a programming language.

Neophyte thinks else than an expert or professionals. What's easy and can be answered in a many seconds by a professional can take a long time and be delicate for a neophyte. neophyte needs to know the syntax, variable, circles, how to reiterate, etc., while professionals have formerly acquired the knowledge. Neophyte generally struggles on how to remedy as they do n't generally know what should be passing. Some adverse circumstances that were indeed noticed at universities like Lappeenranta University of Technology (LUT) where learners of programming failed to pass the introductory course and indeed if they do pass, they were atrocious. Neophyte generally notices that nearly all programming tools expensive and decent Integrated Development surroundings (IDEs) perplexing. So, cheap yet high characteristic tools should be selected Neophyte needs to use small tasks and tools suitable for their standard not for the position of a professional.

To a neophyte, it's burdensome and complicated to learn all the motifs like object- acquainted, dynamic memory running, etc., as they aren't well equipped(4). It's essential to execute an unequivocal and applicable scheme while learning a programming language since the first programming language leaves a lasting print regarding the perception and ambition for learning programming. Learner's attention is divided if they're continuously faced with the new programming terrain. So, it's necessary that they continue to master the terrain they're familiar with analogous abecedarian structure blocks are used by utmost of the programming language, so it'll be easy and royal to pick another new language if one has come complete in a particular language. Indeed though programmers use different languages, they generally incline to use the style and structure of their first language. If applicable way is commenced computer programming is delightful and easy. Therefore, opting a particular programming language is a pivotal step faced by a freshman so as not to be complexed and cheapen their confidence in programming.

Among the colorful languages available, only Java and Python are chosen for this paper. Fashion ability of Programming Language (PYPL) Git- mecca indicator and the fourth Annual inventor Ecosystem Survey by JetBrains was used to get some ideas regarding the current programming trends. Java is one of the most in- demand and popular languages and has been thriving for around 25 times. Python is aged than Java i.e., it has been around for 30 times. Python has been soaring in fashionability for the once many times and suitable to challenge the popular languages. This paper compares colorful characteristics and features of both languages.

### 1.1 Objective

The aspiration of this paper is to dissect and recommend which programming language will be better for a freshman similar that they don't lose interest in the field of programming. This paper doesn't mean to affirm that the programming languages mentioned in this paper surpasses all the other programming language available and is the stylish one for a freshman. opting a language that's freshman-friendly is veritably important for a freshman. newcomers generally don't have any knowledge of which language to choose, so in this paper, we tried our stylish to suggest a language for the neophyte. opting a particular language and sticking to it's a pivotal commitment else the learners will be complexed and it'll disapprove their confidence.

This snap focuses on Python and Java just. Python uses veritably simple English and it's veritably easy to learn. It was erected with the conception of a freshman. While Java is also hands-down to master, it has strong neighborhood reinforcement. The job occasion for both languages is also veritably high. The demand for inventors with the knowledge of these languages keeps on adding . But the demands aren't suitable to be fulfilled as they needed professed inventors so the demand and force chain isn't direct.

The first part of the thesis focuses on the comparison of the syntax of the two languages and observe which bone is less complicated and easy to comprehend. Eventually, the QuickSort algorithm and a simple Tic- Tac- Toe game are enforced in the two mentioned languages.

---

## 2. LITERATURE REVIEW

**Monica,N.,O. Ogbuokiri, Benedette,O. Okwume( 2015)**, offered a report on “ Comparison of python and java for use in commandment in the first program in computer programming ”. This assertion explains the comparison of programming languages( Python and Java) grounded on law size, prosecution time, memory consumption, correctness/ heartiness, and opining trust ability. The job demands and payment for Java and Python inventors are emphasized in this paper. The algorithm for calculation of adding grade point normal( CGPA) was executed and enforced in both the language.

**E Insanudin( 2019)** managed exploration on the comparison of wellspring law perpetration in Python and Java. The bubble kind algorithm is harnessed in this document to dissect the comparison. The comparison is conducted grounded on ranges of law, train capacity, and access speed. The same bubble kind algorithm is enforced in both the language i.e., Java and Python. The author concluded that python has a lower number of canons and lower train capacity than Java and access speed of Python is also good.

**A Bogdanchikov, M Zhaparou, and R Suliyev( 2013)** presented a report on “ Python to learn Programming ”. Python is used for this report as it has neatly organized syntax and important tools to break any task. Python is easy and simple calculation. Some analogous canons are enforced in Python, Java, and C and are anatomized. Python is easy to read and understand so, it's favorable for newcomers. neophyte generally understands programming well when Python is used. The report also presents a result of the quiz marks of the same courses tutored in Java and Python. A hike of 16 is observed in the course tutored in Python.

**Jussi Pekka Kasurinen( 2007)** offered a report on “ Python as a programming vocabulary for the preparatory programming program ”. This snap anatomized the introductory position education and pursue some common problems among the course. Python is compared to C, C, and Java to prove the conclusion that Python is better for the neophyte. The report also anatomized what to include in the course structure and what to educate on the first programming course. Louisiana State University and the State University of West Georgia applied Python to their introductory programming course. The circumstances at fall Lappeenranta University of Technology( LUT) 2005 is also banded in the report. Accumulating good course books and reference material was emphatic.

**Akesson Tobias & Rasmus Horntvedt( 2019)** presented a report on “ Java, Python, and JavaScript, a comparison ”. The report presents a brief comparison of three different languages Python, Java, and JavaScript- grounded on three different phases i.e., its syntax, why the specific language is preferred than another in circumstances, and speed. Python is said to be the easiest among the three languages mentioned which bear for lower time to write. Java's strict syntax is also profitable and favorable. Python is rendered as the laggard from the diverse two vocabularies.

**R. Srinath (2017)** conducted a exploration paper on “ Python – The Fastest Growing Programming Language ” This paper explains why Python is a suitable language for both literacy and real- world programming. Why Python is considered as the fastest growing language is also meditated in this paper. The most important features of python language, the types of programming supported by python and its druggies, and its operations are also examined then. Python also has some cons like the larger and complicated a program is, it's emphatic to maintain and correct crimes that arise; and as Python is stoutly compartmented, the machine needs to perform redundant work making Python slow.

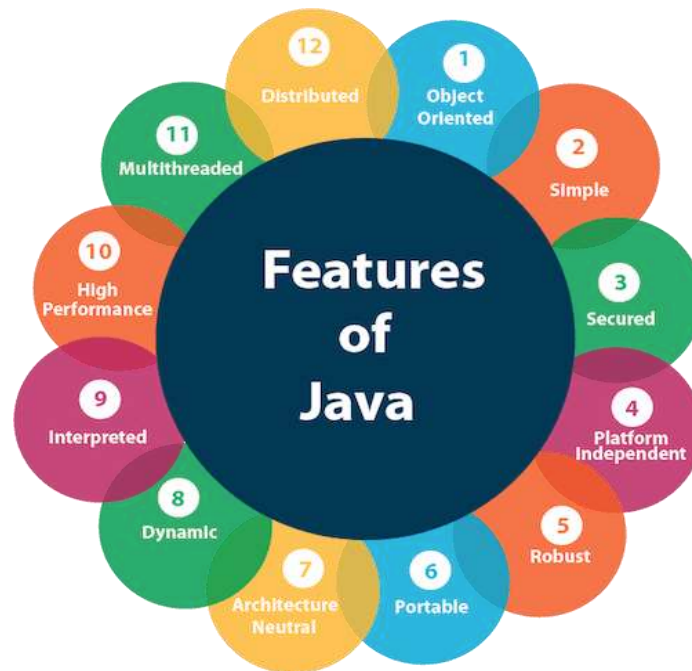
### 3. OVERVIEW

There are colorful languages that may be better for a programmer but in this report, the platoon named only two languages as per their convenience. The named programming languages i.e., Java and Python are well- established and top- ranked in famed websites. They're popular and the demand in the job request is relatively satisfactory. Some of the most important features of a programming language for a freshman should be that it's simple, accessible, believable, and easy to understand. The platoon conducted some exploration concerning the two languages from different sources similar as exploration papers, books, papers, etc. A brief study of both languages is conferred below.

#### 3.1 Overview of Java

The programming language, Java was proposed by James Gosling and his platoon from Sun Microsystems in the time 1991 and was released in 1995( 5). Java's most prominent point is that it's platform-independent i.e., it has the parcels of WORA( Write formerly, Run Anywhere). When Java was first proposed, it was called OAK. OAK was introduced with the perception of a programming language that acts as a platform for connection for appliances like VCR, television,etc.,( 8). Oracle Corporation carried Sun Microsystems in 2009- 10 and came the owner of Java. Java is a collected language that's statically compartmented i.e., their variables are to be declared before assigning values. The programs authored in Java runs hasty than Python but is laggard when equated with C. Airbnb, Uber, LinkedIn, Pinterest, Groupon, Spotify, Eclipse, Hadoop,etc., are substantially grounded on Java. Big Companies like Infosys, TCS, Wipro, HCL Tech, Naukri, Jabong, Myntra, Flipkart, Trivago, ibibo,etc., are still using Java.

#### 3.1.1 Features of Java



**Fig -1:** Features of Java

Figure 1 shown above is a diagram containing some features of Java.

#### 1. Object- acquainted

Java is completely object- acquainted. The OOP (Object- acquainted Programming) helps in dealing with real- world operations. The addition of heritage, polymorphism, abstraction, and encapsulation makes a program an OOP.

#### 2. Platform Independent

When collecting a program, it's collected into a platform-independent byte law which is also executed using a Java Virtual Machine( JVM). The use of JVM makes Java programming platform-independent since if JVM is installed the same program can be executed in multiple platforms.

### 3. Secured

Java uses its own runtime terrain i.e., JVM, Java operations are secure. Safety aspects like Type- checking at collect time and runtime checking are set up inbuilt in Java. Java is also comprehended for the safeguard that it provides. Java also lacks indexes , which encourages protection.

### 4. Robust

Java has strong reminiscence operation and it automatically gets rid of objects that aren't harnessed. Java consists of exception running and kidneys of consisting ways. All these features of Java make it well-conditioned.

### 5. Movable

Java byte law can be transferred to any platform without any perpetration making it movable **6. Multi-thread**

The features of multi-threading are inbuilt in Java. It aids in erecting largely interactive and responsive operations that deal with numerous tasks at formerly. Multi vestments partake a common memory area, increase the capabilities and performance.

### 7. Distributed

This aspect of Java allows penetrating lines by calling the styles from any machine on the internet. It supports the sharing of data and programs among multiple computers for networking that's naturally integrated into it. Java supports RMI( Remote Method Invocation), Socket Programming, and the COBRA that aid in participating objects in a distributed terrain.

#### 3.1.2 Applications of Java

There are many fields that use Java in the real world. The team conducted some research regarding the applications of Java.



Fig -2: Application of Java

A brief definition of some of the frequent uses of Java is offered below.

#### 1. Android Applications

Java is regarded as the sanctioned programming language for android mobile app development. Although mobile operations can be created using Dart, Java is substantially preferred. The Java byte law collected runs on a technical virtual machine for android called Dalvik Virtual Machine( DVM). operation creating software like Android Studio and Kotlin is compatible with Java. Java OOP principle gives better security, simple and more effective with developing android operations

#### 2. Desktop GUI Applications

Java can be used for the developed desktop operation. The packages similar as Abstract Windowing Toolkit (AWT), JavaFX, and Swing are used to make GUI operations.

### 3. Web- grounded Applications

Java is used to produce web operations with the help of servlets, struts, JSP( Java Garçon Pages), etc. Java caters easy coding and high security which facilitates the development of operations for health, social security, education, and insurance. Open- sourcee-commerce platforms like Broadleaf give aid to Java in developinge-commerce web operations.

### 4. Pall- grounded operations

Pall computing gives a low- cost result for IT structure. It presents with on- demand delivery of IT coffers through the internet comprising of storehouse, waiters, database, networking, and software with the pay- as- you- go pricing model. Java has characters that aid in development like SaaS( Software- as-a-Service), IaaS( structure- as-a-Service), and PaaS( Platform- as-a-Service).

### 5. Big Data Technologies

For the study of Big Data, Java is used as it's presto, dependable, and robust. The features of Java similar as Automatic Garbage Collection and strong memory operation make it favorable for use in Big Data. fabrics like Apache Mahout, Apache Spark, Java JFreechart are used by Java for engaging in Big Data. Java is abysmal- seated with Open source neighborhoods making it more commendatory for all these technologies. Big Data Technologies like Apache Hadoop, Apache Spark, Apache Mahout,etc., aresub-projects of Java.

#### 3.1.3 Advantages of Java

1. Java is forthright, explosively- compartmented, and has strict prospects that govern the learners to contemplate in the correct way.
2. Java is easy to use, write, collect, remedy, and learn compared to other languages like C, C,C#.
3. Being Platform-independent, distributed language, supportingmulti-threading, furnishing Automatic Garbage Collection,etc., are some of the boons of Java.
4. The use of OOP in Java enables to set up standard programs andre-useable law.
5. Java has an inbuilt rich API for tasks like database connection, networking, I/ O, XML parsing,etc.
6. Java was developed with consideration of a secured platform. It has a Security director for each and every operation where the access rules for classes are established.
7. Java has a vast number of well- tested libraries and fabrics.
8. Java is fairly affordable to maintain as it doesn't count on an unequivocal tackle frame.
9. Another intriguing aspect of Java is that it has strong community support. So, there will always be someone who's suitable to help when the learner gets wedged.
10. Although, Java is analogous to C and C, Java doesn't include characteristics similar as pointers and multiple- heritage.

#### 3.1.4 Disadvantages of Java

1. Starting from 2019, Oracle blazoned that Java Standard Edition 8 will be charged for business, marketable, and product.
2. Java particularly focuses on storehouse and not on the backup of data.
3. Memory operation is expensive as large memory space is needed.
4. Java is slower and memory consuming when compared with C or C.
5. Opting a tool to develop a GUI is tough as Java absolutely lags when in desktop programs.
6. Java canons are circumlocutory. It fixates on being more manageable but jeopardized with immensely perplexing canons and lengthy information.

### 3.2. Overview of Python

Python was formulated in the late 1980s and executed in December 1989 by Guido van Rossum at Central Wiskunde & Informatica( CWI) in the Netherlands( 5, 7,16. Python was proposed to be the heir at law for the ABC language that's complete for exception running and interfacing with the operating system Amoeba( 7, 17). Python was named corresponding to Guido's passion for the TV show Monty Python's Flying Circus( 2, 17). Python is interpreted and stoutly- compartmented programming language which means that programmer doesn't need to define the data- type of the variables and no need for compendium and with the use of the interactive command- line, they get prompt assessment without having to stay for the whole program to be finished. Python Software Foundation( PSF) is anon-profitable association rooted as the intellectual proprietor of Python since interpretation2.1( 2). Python has come the swift- growing language. The fashionability of Python in data wisdom is one of the main reasons for the hike of Python( 9). Some

software programs that are authored in Python are YouTube, Google, Instagram, Reddit, Spotify, Dropbox, Quora, etc. troupes like IBM, Disney, NASA, Instagram, Spotify, Amazon, SurveyMonkey, Facebook, etc., application Python.

### 3.2.1 Features of Python

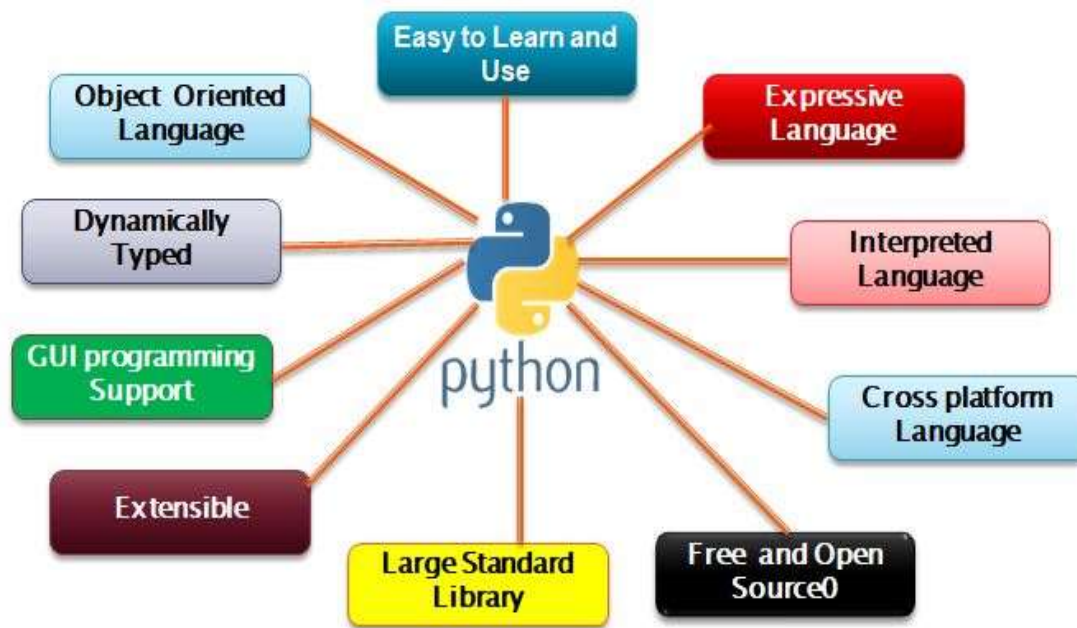


Fig -3: Features of Python

Figure 3 shown above are some common features of Python.

#### 1. Easy

Python is easy to code and easy to read as compared to other languages like Java, C, C++, etc. Python syntax can be studied by anyone during a short period of time. Python code is like English that allows the learner to focus on the result.

#### 2. Expressive

Python can execute a complicated function with only a few lines of code compared to other languages.

#### 3. Free and Open Source

Python is open source and freely available. The public can assist and contribute to the improvement of the language. The Python source code can be downloaded, changed, used, and be distributed.

#### 4. High-level language

Python is a high-level language. There is no need for remembering the architecture and memory management which makes python very favorable.

#### 5. Portable

Python is said to be portable as the same program written in Python can be executed in different platforms such as Windows, Linux, Unix, etc., if system-dependent features are avoided.

#### 6. Interpreted Language

Python is an interpreted language. The code does not need compilation, they are executed line by line and not all at a time which makes debugging the code easier than all the other languages. So, Python is slower than Java due to this feature.

#### 7. Object-Oriented

Python supports an object-oriented approach which helps the programmer to write reusable code and aid in developing the application with lesser code.

#### 8. Extensible and embedded

The extensible property of Python allows code to be written and compiled in other languages like C or C++. This code then can be used further in Python when required. The embedded property of Python allows is allowing the use of Python in another programming language.

### 9. Large Standard Library

Python along with the large standard library provides for a large range of modules and functions. So that the programmer does not need to write the code, they can just import it

### 10. Dynamically Typed

Python is said to be a dynamically typed language because it does not need to specify the data-type of the variable while declaring it. The type of value is decided during the run time.

### 3.2.2 Applications of Python

There are many uses of Python but the team only selected some common applications.



**Fig -4:** Application of Python

The applications of Python are shown in brief below.

#### 1. Web development

Python is a go-to language for web development. Django, Pyramid, Flask, Bottle are some of the frameworks offered by Python. Python web frameworks are popular for their security, scalability, and flexibility. Requests, BeautifulSoup, Paramiko, Feedparser, Twisted Python, etc., are libraries that also included in the Python's Package Index.

#### 2. Game development

Python has many in-built libraries that are favorable for developing a game. PyGame, PyKyras are frameworks for game development and PySoy is a 3D cloud game engine for Python3.

3. **Artificial Intelligence and Machine Learning** Artificial Intelligence and Machine Learning is one of the rising topics and will be continued in the future. Python is popular and favorable to be used in Artificial intelligence and machine learning due to its character of being stable, secure, flexible, and of its various tools. Some of Python libraries and frameworks used in Artificial Intelligence are SciPy, Pandas, Seaborn, Keras, TensorFlow, Scikit-learn, NLTK, Pytorch, Accord.NET, etc.

#### 4. Desktop GUI applications

Python is also used for desktop applications. GUI toolkits and frameworks such as PyQt, PyGtk, Kivy, Tkinter, WxPython, PyGUI, and PySide help in making the development of eminently functional desktop application an easy task.

#### 5. Web scraping applications

A Python is a marvelous tool that can be used to excerpt large data from a website which is then used for job listings, price comparison, etc. BeautifulSoup, Mechanical Soup, LXML, etc., are some tools used for web scraping.

## 6. Data Science and Data Visualization

Python is preferred by many for the analysis and visualization of large data. Python is associated with statistical means to analyze and depict complicated data by data scientists. Packages such as NumPy, Pandas, Sci-Kit, etc., are used.

### 3.2.3 Advantages of Python

1. Python is easy to read, learn, and write. So, it is beginner-friendly.
2. Lesser code is required by Python compared to other languages for the same task.
3. Python is free and open-source. Thus, it is broadly used for varied functions.
4. Python has large community support.
5. Python is dynamically typed, embedded language.
6. Due to its vast libraries the programmer can execute complex functions easily.
7. Python is an eminently flexible programming language.
8. Python is enormous for data visualization making the reports and visual presentation of data easy to understand.
9. Python provides a low learning curve due to it being simple and easy and is a productive language.
10. Python being an interpreted language executes the code line by line. So, even if there are multiple errors only one error will be shown at a time without further execution when an error occurs.
11. Python offers multiple programming paradigms like object-oriented programming, imperative and functional programming, structured programming, procedural programming [9]. Unlike Java, Python supports multiple inheritances.

### 4.2.4 Disadvantages of Python

1. As Python is an interpreted language, it is slower in execution when compared with other languages.
2. Python is not favorable for mobile development.
3. Python is not preferred for memory comprehensive task.
4. As Python is dynamically typed raises run time error leading to restriction in design.
5. Python is used somewhat less in large companies and businesses due to its limitation in database access layers unlike JDBC (Java DataBase Connectivity) and ODBC (Open DataBase Connectivity).
6. The Global Interpreter Lock (GIL) of Python allows the execution of only one thread at a time.
7. Python's simplicity becomes a disadvantage when the programmer wants to shift to other languages like Java which have strict structure.

---

## 4. Method

The report for a comparison of the two languages Java and Python is portrayed in this thesis. The comparison is processed through two different paths. Java 8 is used in EditPlus and Eclipse IDE to write all the Java code and Python 3.7.6 is used to write Python code in Jupyter Notebook. The first part of this thesis generally fixates on the syntax comparison of Python and Java. Here, the team resolved to select some basic common syntax and features for learning a programming language. The elements included in consideration of the report are shown in figure 3. The second part of the thesis is an implementation of QuickSort and a game called Tic-Tac-Toe. They are implemented in both Java and Python and the comparison was done using the Lines of code (LOC), file capacity, and speed.



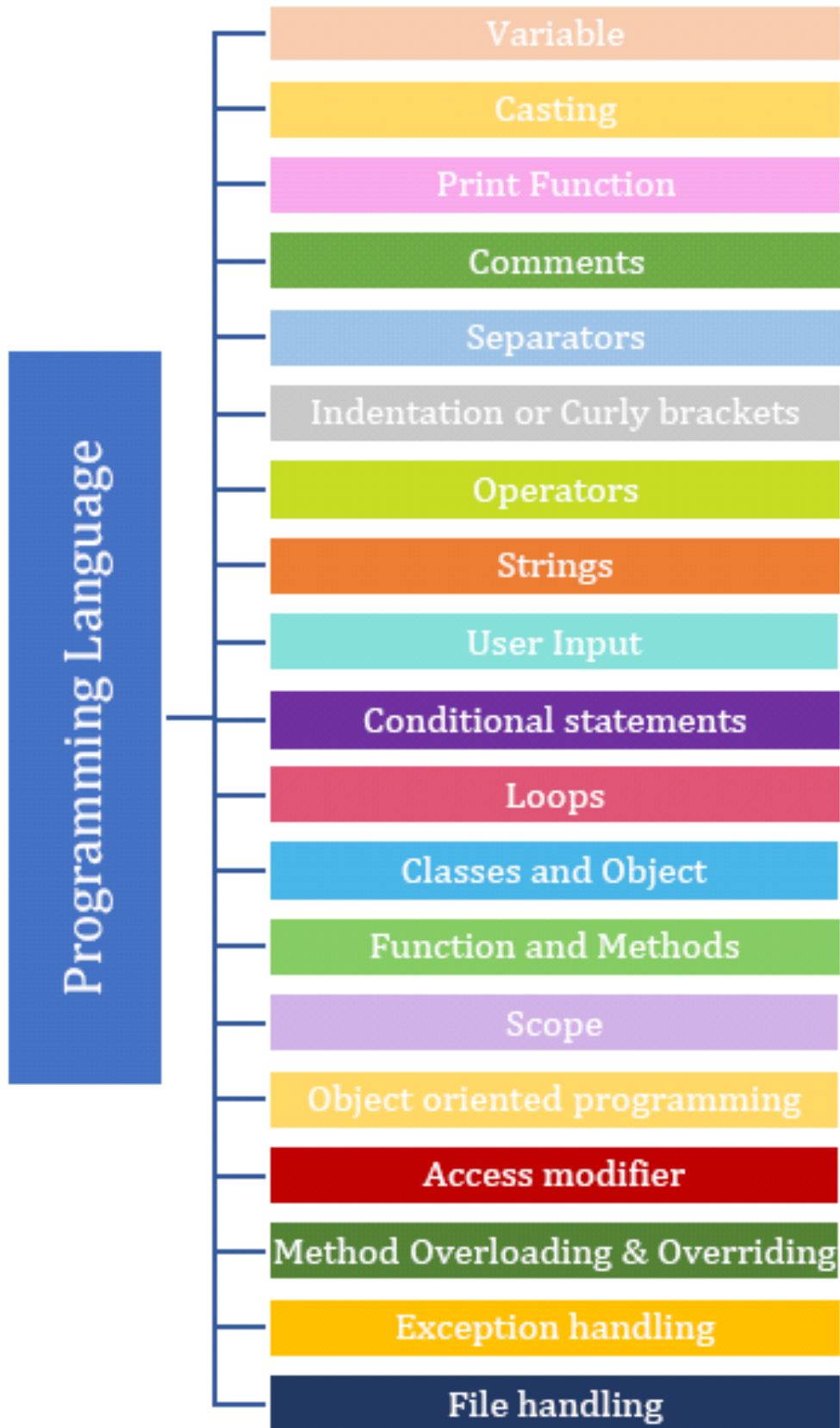


Fig -5: List of the considered elements.

## 5. Result and Discussion

### 5.1 Syntax comparison between Java and Python

The team conducted an analysis of all the topics mentioned in above figure3. The findings and differences are written below after the comparison was conducted. Due to the limitations of time the team was not able to do the study in many details. There were many important features that were not able to be mentioned in this report due to the restraint amount of time. Appendix A contains all the figured mentioned in this section.

#### 1. Variables

Variables in Python (shown in figure A1) need not be declared with a particular datatype before assigning the value. This makes Python dynamically typed. The values once set can be changed to another different type, i.e., if a variable  $x = 5$  where 5 is an integer, in the next step it can be changed to  $x = 1j$  where  $1j$  is a complex number. Multiple variables can be declared and stored with the same value in one line using “=” or with different values respectively separated by a ‘,’. Python has 2 types of variables: class variable (static variable) and instance variable which will be discussed later.

Variables in Java (shown in figure A2) however need to be declared beforehand along with its type. Although a variable can be declared before without assigning the value, it must nonetheless affirm its datatype while declaring the variable. A variable’s value can be overwritten but it must be of the same datatype. Java has a keyword called “final” which helps in preventing the value from being replaced. Many variables of the same type can be declared using comma-separated list. Java has 3 types of variables: local variable, static variable (class variable), and instance variable.

#### 2. Casting

Casting is the process to specify a data type to a variable. In Python, the “type ()” function is used to display the type of the value stored in the variable. Casting can be done during the time of declaring the variable like  $x = \text{int}(1.8)$  i.e.,  $x = 1$  where the  $\text{type}(x)$  will give an int. Another method of casting can be done after the value has been assigned such as  $b = \text{str}(a)$ , where  $a = 1$ , which means convert the integer 1 stored in a to a string. Python type casting is shown in figure A3.

Java has two types of casting: -

a) implicit or widening or automatic type casting:

It is the process of converting smaller data type to a larger data type like byte to short or int to double. If the conversion is from short to int where variable a is a short then  $\text{int } b = a$ ;

b) explicit or narrowing or manual typecasting:

It is the process of converting larger data type to a smaller data type like double to short or int to byte. If the conversion is from float to int where variable x is a float, then  $\text{int } y = (\text{int}) x$ ;. Java type casting is shown in figure A4.

#### 3. Print Function

Python only uses `print ()` to display the output (shown in figure A5). There are two ways to concatenate a variable to a print statement. Python uses “,” to display variable in the print function. Whitespace is added automatically before the value of the variable. To concatenate a variable to a print function i.e., using “+” character then the data type of that variable must always be a String.

Java has a more structured syntax. To concatenate a variable in a print statement “+” is used. `System.out.print();` prints the strings or references and `System.out.println();` is also similar to the above but after printing it moves the cursor to the next line. Java print function is shown in figure A6.

#### 4. Comments

Comments are text or code that is written in the source code to make it more readable. They are not executed. Python uses the “#” character to declare it as a comment. For multi-line comments “#” can be used for each line or “`“ “ “ “ “`” (triple quotes) can be used as python ignore string literals which are not assigned to a variable (shown in figure A7).

Java uses “`//`” (two forward slashes) character for single-line comment, everything after the “`//`” is regarded as a comment. For multi-line comment, “`/*`” is used at the start and “`*/`” is used at the end of the comment (shown in figure A8). Any text or code between them is ignored.

#### 5. Separators

Python uses whitespace and does not have a specific symbol to declare a line break (shown in figure A9).

6. Java uses semicolon “`;`” character to separate two lines from each other (shown in figure A10). After every statement if “`;`” is not used, it will raise an error. Programmers are emphasized to check the “`;`”, as a massive error will arise even if a single “`;`” is missed.

#### 7. Indentation or Curly brackets

Python uses whitespace to indicate if the code is inside a loop or not (shown in figure A11). If the codes are inside a loop it is indented and to indicate code outside the loop it is outdented. The whitespace indentation makes the code readable. If there is a mistake in the indentation then an error will arise.

Java uses curly brackets to determine if the code is inside a function or loop, etc., (shown in figure A12) . If the code is inside curly brackets then it means it is within the function. It does not need to follow the indentation rule which makes their code messy to read.

## 8. Operators

Python has many forms of operators such as arithmetic operator (+, \*), assignment operator (=, +=), comparison operator (=, <), the logical operator (and, not), identity operator (is, is not), membership operator (in, not in), the bitwise operator (^, |) (shown in figure A13).

Java also has different types of operators, some of them are also similar to Python. Java has arithmetic operation like ++ and -- for increment and decrement, unlike Python. Java does not have an identity and membership operator. For a logical operator in Java &&, ||, ! are used for and, or, not respectively (shown in figure A14).

## 9. Strings

In Python, strings can be written using a single or double quotation (shown in figure A15). There is no character datatype in Python. Multiline strings can also be declared using triple quotes. In Python, strings are array and there is negative indexing in Python. Strings in Python are also immutable. There are many built-in methods for the string. The format method in string helps in combining string and number without type-casting. An escape character '\ ' if an illegal character inside a string is needed.

Java string can be declared using double quotation "" (shown in figure A16). Java has a separate datatype called character datatype assigned with a single quotation '. Strings are treated as objects in Java. Strings are immutable in Java but can be made mutable using StringBuffer and StringBuilder. Java also has an escape character like Python. There are built-in methods for a string in Java. Multiline string literals are not supported in Java.

## 10. User Input

In Python, the keyword "input" is used for the user to input value (shown in figure A17). Everything a user input Python recognizes it as a string data type. So, casting becomes necessary if a specific data type value is needed from the user.

Getting input from the user is not as simple in Java as in Python. The Scanner class object is created and java.util.Scanner is imported so that user can input values. "nextInt()", "nextFloat()", "nextLong()", "nextDouble()", "next().charAt(0)" methods are used to get integer, float, long, double, character inputs respectively from the user (shown in figure A18, A19). "next ()" or "nextLine()" is used to get a string.

## 11. Conditional statements

A conditional statement allows a certain group of code to execute according to the true or false of the given condition. Python has if, else, elif and nested if condition (shown in figure A20). It does not include switch-case like other programming language and "elif" is used instead of "else if". As Python relies on indentation, in order to specify that the code belongs to the specific condition, the block of code must be indented and after the condition, there must be a ":" character. Ternary operators or shorthand methods for if or if-else is present in Python which is used when there is only one statement to be executed and it is put in the same line.

Java's conditional statements include if, else, else if, switch case (shown in figure A21). Both the language, Python and Java use the following logical conditions <, >, <=, >=, ==, !=. The condition must be inside a "()" bracket and the statements or code to be executed inside a "{}" bracket. Java has shorthand if-else to replace simple if-else condition.

## 12. Loops

Python has while loop, for loop and nested loops (shown in figure A22). There is no do-while loop in Python. For loop in Python doesn't require to be indexed beforehand, it is more of an iterator method. Unlike other programming languages, Python for loop can have an "else" method which is to be executed when the for loop is over. There can be nested for loops. While loop in Python is executed as long as the condition is true. In Python, variable in while loop is to be initialized before the loop and incremented inside the loop. There can be nested while loops. Python while loop can have else method after the completion of the loop.

## 13. Classes and Objects

Classes are logical grouping which helps in the reuse of data as part of a code. An object is a collection of data and methods and class is the blueprint of that object. The role of an object is represented by attributes (variable) and action by the method. Python is an object-oriented programming language and entirely everything is an object. Python classes are declared using the keyword "class" (shown in figure A24). In the example, "s" is the object (instance) of the class "Stu". "self" refers to the current class instance. It used to access variables that belong to the class. The memory location of the instance that is created is internally passed to "self". Python also has a special constructor method which will be discussed later. Variables are of two types in a class i.e., class variable (or static variable) and instance variable. The class variable is shared by all instances of a class whereas instance variables are variables that create a separate copy of the variable in every instance. Class variables can be accessed from any method within the class. The instance variable requires an instance method to access it.

## 14. Function and Methods

Python function is a block of code that is executed when called. There are built-in functions and user-defined functions. To indicate as a function "def" keyword is used (shown in figure A26). If there is a function say, "def func(x):" then it can be called using "func(a)". Here, x is a parameter and a is an

argument. There is also a term called arbitrary argument denoted by `**args` if the number of arguments to be passed is not specified. Python also supports keyword arguments and the order of the argument doesn't matter here. Every argument passed to a function is treated as the same datatype. Pass statement can be used to avoid error if the function is empty. If there is no return statement in a function then "None" is returned. The function or method in Python supports multiple return values simultaneously. A function written inside a class is called a method (shown in figure A27). Python has a special method called constructor i.e., `def __init__(self, parameter):`. It is used to declare and initialize an instance variable. The first parameter for constructor and instance method is "self". The instance method in Python is obligated to object and act on instance variables. Class methods act on class variables and are bound to the class which doesn't require class instance creation. `@classmethod` is used for a class method. The static method is used when some operation is associated with class but does not need the class or instance to process. `@staticmethod` is used for static methods. The team won't dive into details on this topic.

### 15. Scope

The scope is the region where a variable is accessible and that region is where the variable is created. In Python, a variable that is created inside a function or method is called a local variable, and the region is called local scope. A local variable cannot be accessed or modified outside the function. A variable created outside any function is known as a global variable and it comes under the Global scope. This variable can be accessed from any part of the program even inside a function also. The global keyword is used if a global variable is to be modified inside a function (shown in figure A30).

### 16. Object-oriented programming (OOP)

Object-oriented programming is a programming paradigm that is based on the object containing data members and methods. It increases the flexibility and maintainability of programs by making the data member and methods come together. Class is a blueprint for the object. Class and object will be discussed later. The four main characteristics of object-oriented programming are abstraction, encapsulation, inheritance, and polymorphism.

#### a) Abstraction

Hiding certain details and showing only essential information to the user is an abstraction of data. In Python, the abstract method is declared but doesn't have any implementation (shown in figure A32). A class having one or more abstract methods is called an abstract class. Abstract class can't be instantiated and to implement the abstract method sub-class is required. Python by default doesn't provide abstract class, so it comes with a module named ABC (Abstract Base Classes) and uses the keyword `@abstractmethod` as a decorator to make a method abstract.

In Java, a class declared with the "abstract" keyword is called abstract class (shown in figure A33). It can have abstract and non-abstract methods. An abstract method cannot be instantiated (an object cannot be created) but can have a constructor, static, and final method. A method is abstract if it is declared with the keyword "abstract" and doesn't have an implementation (i.e., no method body). The abstract method must always be followed with a ";". The Abstract method can only be used with an abstract class in Java.

#### b) Inheritance

Inheritance is the process of creating new classes from existing classes so that the new class can acquire all the features of the existing class. It provides code reusability. In Python, the new class created is called the child class or superclass and the class from which it is inherited is called parent or subclass. In Python, a base class can be inherited by a derived class by just mentioning the base class in the bracket (shown in figure A34). The child constructor overrides the inheritance of the parent constructor (i.e., constructor overriding) so, to keep the parent constructor add a call of the parent's "init" function. The `super()` is used to call the parent class constructor or method from the child class (shown in figure A35). The `super()` function helps the child class to automatically inherit all the properties and methods of the parent class. Python supports single inheritance, multiple inheritances, multilevel inheritance, hierarchical inheritance, and hybrid inheritance.

#### c) Polymorphism

Polymorphism is one of the important features of object-oriented programming which allows a single action to be performed in different ways. If a class contains the same method with different implementation then the class is polymorphic. Polymorphism makes code easy to change, maintain, and increase flexibility. In Python, operator overloading where an operator performs additional action is an example of polymorphism. Functions such as `len()` that can be executed with multiple data types are polymorphic. Python allows duck typing where the object that is not checked while invoking the method on the object and any object is accepted if the method is found in the object (shown in figure A38). Method overriding where the subclass method replaces and overrides the same method in the superclass during inheritance is also an example of polymorphism.

#### d) Encapsulation

Encapsulation is wrapping up of data under a single unit. It is a protective shield that prevents unauthorized access to data from outside the shield. Access modifiers which will be discussed later play an important role in data encapsulation. Python has a public, protected, and private members (shown in figure A40). Java has public, private, default, and protected members (shown in figure A41). Encapsulation makes our code cleaner and easy to read. Encapsulation is a way to achieve data hiding in Java so that other classes will not be able to access data in private data members. In Java, by providing only the getter and setter method we can make the class write-only or read-only. If all the variables in the class are declared private and the getter and setter method public to set and get values of variable then it is considered as encapsulation.

### 17. Access modifier

Like all the object-oriented programming languages, Python also has access modifiers i.e., private, public, protected. The access modifier helps in preventing the data from being exploited and secure it from unauthorized access. With the help of the underscore “\_”, Python specifies access modifier. The members declared as public are accessible from outside the class by everyone. The members declared as protected are accessible from within and outside the class but only in a class derived from i.e., a child or subclass. Prefix single underscore “\_” being added to a data member of a class make it protected (shown in figure A42). The private members are only accessible within the class and the member for a class are made private by adding double underscore “\_\_” (shown in figure A43). All members in a python program are by default public.

### 18. Method Overloading & Overriding

Method overloading occurs when there is more than one method with the same name but different parameters in the class. Method overriding is a concept of replacing the same method with the same number of parameters in the child class that is defined in the parent class. Both method overloading and overriding are features of polymorphism. Method overloading is not supported in Python (shown in figure A46) because Python only keeps the last declared method. Python supports method overriding (shown in figure A47). Python constructor supports overriding only.

### 19. Exception handling

Exception handling is the process done by the programmer to handle the run-time errors leading to termination of the normal execution of a program due to erroneous input by the user, network connection problems, etc. There are many in-built exceptions in Python that are raised when the Python interpreter encounters an exception. The code or operation which can raise an error are placed in the “try” block, the exceptions are then handled in the “except” block. The “finally” block is used when there are certain codes that need to be executed regardless of the exception being handled or not (shown in figure A50). Python also has specific exceptions. If the exception is not executed and else block is also present only then else block will be executed. Python also can throw an exception by using the “raise” keyword.

### 20. File handling

Python has an inbuilt method for reading, writing, and creating files so importing an external package is not needed. To open a specific file in a specific way “open ()” function along with the mode as a parameter is used. The different modes of operations are r (read-only) it arises an error if the file does not exists, w(write) it creates a new file if it does not exist, a(append to the end of the file) it create a new file if it does not exist, x(write) create the specified file and return an error if the file exists, t(text mode), b(binary mode), +(open a file for updating) (shown in figure A52). The OS module is imported and “os.remove()” is used to delete a file (shown in figure A53).

## 5.2 Code Implementation

The team implemented the Quick Sort algorithm and Tic-Tac-Toe game in both Java and Python and also analyzed the result in terms of Lines of code, file capacity, and speed.

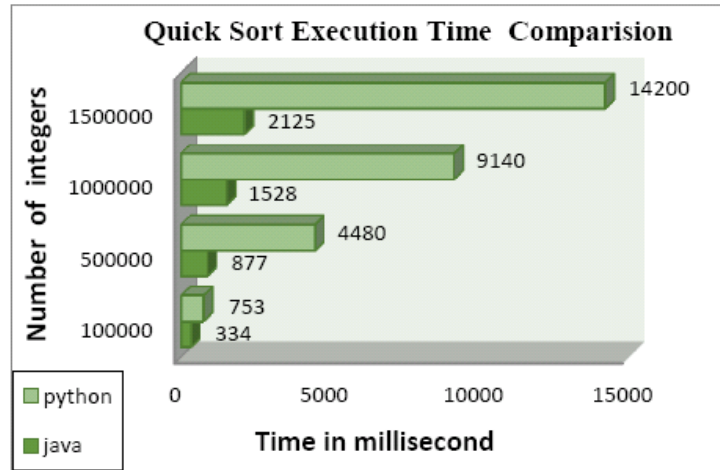
### a) Quick Sort Algorithm

Only the code for the QuickSort algorithm was from the GeeksforGeeks website so as to achieve more reliability in the structure of the algorithm. The team added additional code for reading text files that contain the numbers to be sorted and run-time calculations. The team created four text files containing 100000, 500000, 1000000, 1500000 randomly generated numbers respectively. The sorting algorithm was then executed using each text file for both the languages. For Python, the “%%timeit” Jupyter magic function is used to calculate the approx. execution time. For Java, “Instant.now()” is used to calculate the elapsed time. Python consumed more time than Java for the program to be executed in all the four cases. Since the algorithm is for sorting, the program in both the language executes faster when executed a second time.

From Table 1, it is observed that Java is faster than Python, and Figure 4 displays the graph for the comparison.

**Table 1:** Quick Sort Execution Time Comparison

Number of Integers to be Sorted	JAVA	PYTHON
1,00,000	334ms	753ms
5,00,000	877ms	4480ms
10,00,000	1528ms	9140ms
15,00,000	2125ms	14200ms

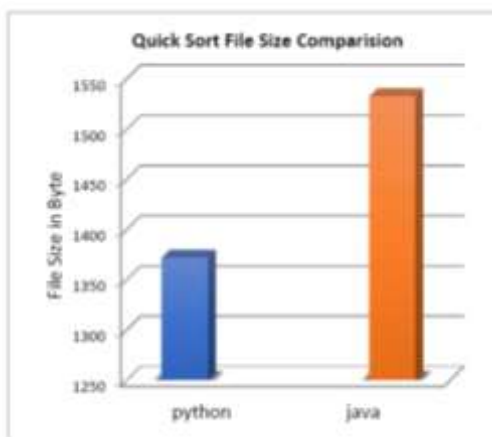


**Fig-4:** Quick Sort Execution Time Comparison Graph

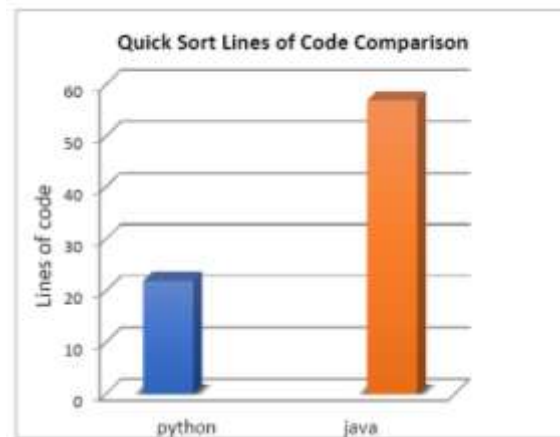
From Table 2, it is observed that Python has lesser lines of code and file size when compared with Java. Figure 5 and Figure 6 displays the graph for the comparison.

**Table 2:** File Size and Lines of Code Comparison in Quick Sort

Programming Language	Lines of Code (LOC)	File Size (in bytes)
Python	22	1372
Java	57	1533



**Fig-5:** Quick Sort File Size Comparison



**Fig-6:** Quick Sort Lines of Code Comparison

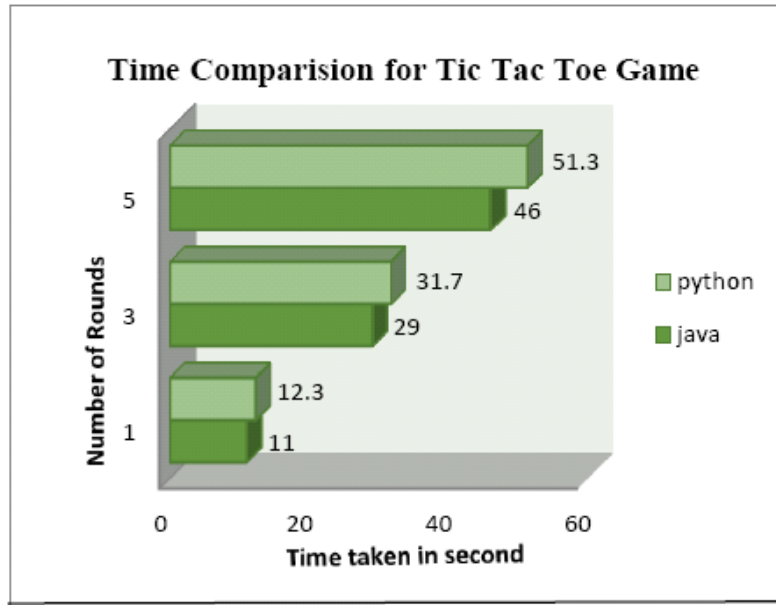
**b) Tic-Tac-Toe Game**

The team implemented the game tic-tac-toe for two players. There is an option for the number of times that the players want to play the game i.e., a game for one round only or three rounds or five rounds. The players input the name they want to be represented and randomly a player will be chosen using the random method in both the languages i.e., Java and Python. According to the randomly chosen player, markers will be inserted respectively. The board of the game has 9 positions and the game will be continued until and unless there is a win or tie. For the game of three or five rounds, the player name and the chosen marker will not be changed and will be flipped together until the game is over. The score will be displayed for every round and the final score along with the execution time is displayed after the game is over completely. The execution time differs according to the time taken by the players to input the value. The team only considered the play where the player wins and the input is given at a somewhat fast speed to analyze the execution time.

From Table 3, it is also observed that Java is faster than Python and Figure 7 displays the graph for the comparison

**Table 3:** Tic-Tac-Toe Execution Time Comparison

Number of Rounds	Python	Java
1	12.3sec	11sec
3	31.7sec	29sec
5	51.3sec	46sec

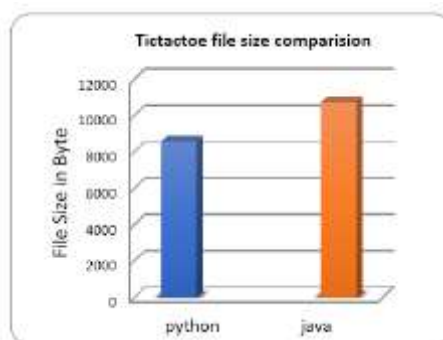


**Fig-7:** Tic-Tac-Toe Time Comparison

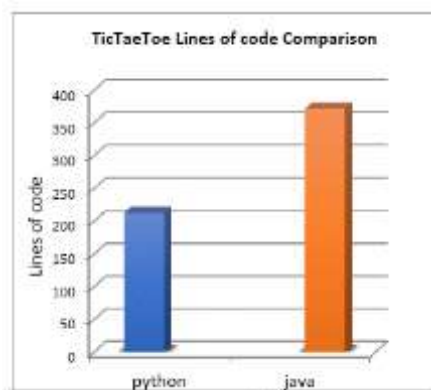
From Table 4, it is also observed that Python has lesser lines of code and file size when compared with Java. Figure 8 and Figure 9 displays the graph for the comparison.

**Table 4:** File Size and Lines of Code Comparison in Tic-Tac-Toe

Programming Language	Lines of Code (LOC)	File Size (in bytes)
Python	213	10729
Java	372	8583



**Fig-8** Tic-Tac-toe File Size Comparison



**Fig-9 Tic-Tac-Toe Lines of code Comparison**

## 6. FUTURE SCOPE

Due to certain limitations where the major debit was a time restraint, the platoon failed to consider further than two languages. Only some introductory syntax and features are considered in this report. There are numerous important characteristics that need to be considered which weren't included in this report. So, it'll be absorbing to compare different other languages along with a deep analysis of the language. Although Python is slow, why it's favored for use in Big Data, Machine literacy, Artificial Intelligence, etc. can also be banded. An trial can be conducted where some beginners are made to suffer a course in the language and observe their performance. Factors similar as security, trustability, continuity, and maintainability may be considered during the comparison. Comparison of language on the base of their uses and which would be favored for the same operations as which language is better to be used for web development can be an ideal content for farther exploration.

## 7. CONCLUSIONS

The platoon conducted an analysis for which programming will be better for a freshman similar that they don't lose interest and confidence in the process. The platoon named only two languages i.e., Java and Python which may not be the stylish option for some cases. The platoon acknowledges the fact that there are numerous other programming languages which surpasses the two mentioned languages. Python and Java were preferred for the fact that they're popular and there's a vast community that supports them. The vast community for these two languages plays an important part since when a freshman encounters an handicap or doesn't understand a conception, they can seek help from the separate communities.

The platoon inferred that both the languages considered in this report have their own advantages and disadvantages in their separate fields. It can not be said with conviction that one of them is better than the other. Although Java is kindly complex than Python in structure, it provides a better understanding of memory operation and is more secure. Python is short, simple, and easy. A neophyte can fluently understand a Python program for it's written in simple English. In Python since indentation is mandatory, it makes the law more readable. still, this isn't the case for Java as there's no effect for indentation and the whole program can be written in one line to make it look short. The use of semicolon which indicates the end of the line in Java is occasionally overlooked which leads to a major compendium error. Python being stoutly compartmented leads to longer prosecution time as the variable type is checked during run time whereas Java is statically compartmented so the exact datatype for variables is known during compendium leading to briskly prosecution than Python.

Utmost of the programming language has analogous fundamentals, thus anyhow of which language is chosen the person can learn another programming language fluently. The neophyte mustn't keep changing the language before learning it as it'll lead to a loss of confidence. So, the neophyte should choose the programming language according to the pretensions they prefer. However, also choose Java or Swift or Flutter, If the person would like to go for app development. However, Java, C or C may be preferred, If the person wants to produce a game than a language like JavaScript. If the person is interested in web development or

Ruby will be favorable. Therefore, a freshman must instead endure what he she would like to engender on in the offing or pick an demesne of claim. After that, checking to their interests, the language must be handpicked.

## 8. REFERENCES

- [1]. Monica, N., O. Ogbuokiri Blessing, and O. Okwume Benedette. "Comparison of python and java for use in instruction in first course in computer programming."
- [2]. Insanudin, E. "Implementation of python source code comparison results with Java using bubble sort method." *Journal of Physics: Conference Series*. Vol. 1280. No. 3. IOP Publishing, 2019. doi:10.1088/1742-6596/1280/3/032027



- 
- [3]. Bogdanchikov, A., M. Zhaparov, and R. Suliyev. "Python to learn programming." *Journal of Physics: Conference Series*. Vol. 423. No. 1. IOP Publishing, 2013. doi:10.1088/1742-6596/423/1/012027
- [4]. Kasurinen, Jussi. "Python as a programming language for the introductory programming courses." (2007).
- [5]. Åkesson, Tobias, and Rasmus Horntvedt. "Java, Python and Javascript, a comparison." (2019).
- [6]. Pellet, Jean-Philippe, Amaury Dame, and Gabriel Parriaux. "How beginner-friendly is a programming language? A short analysis based on Java and Python examples." (2019).
- [7]. Adawadkar, Kalyani. "Python Programming-Applications and Future." *International Journal of Advanced Engineering and Research Development*. [http://ijaerd.com/papers/special\\_papers/IT032.pdf](http://ijaerd.com/papers/special_papers/IT032.pdf) (2017).
- [8]. Fatima, N., and S. Arabia. "Performance comparison of most common high level programming languages." *International Journal of Computing Academic Research (IJCAR)* 5.5 (2016): 246-258.
- [9]. Srinath, K. R. "Python–The Fastest Growing Programming Language." *International Research Journal of Engineering and Technology (IRJET)* 4.12 (2017): 354-357.
- [11]. Pears, Arnold, Stephen Seidman, Lauri Malmi, Linda Mannila, Elizabeth Adams, Jens Bennedsen, Marie Devlin, and James Paterson. "A survey of literature on the teaching of introductory programming." In *Working group reports on ITiCSE on Innovation and technology in computer science education*, pp. 204-223. 2007.
- [12]. Foster, Elvis. "A comparative analysis of the C++, Java, and Python Languages." (2014)
- [13]. McMaster, Kirby, et al. "Java vs. Python coverage of introductory programming concepts: a textbook analysis." *Information Systems Education Journal* 15.3 (2017): 4.
- [14]. <https://www.jetbrains.com/lp/devecosystem-2020/>
- [15]. <http://pypl.github.io/PYPL.html>
- [16]. <https://octoverse.github.com/>
- [17]. [https://en.wikipedia.org/wiki/Python\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))
- [18]. Donaldson, Toby. "Python as a first programming language for everyone." *Western Canadian Conference on Computing Education*. Vol. 232. 2003.
- [19]. Dr. R. Nageswara Rao. "Core Python Programming".
- [20]. New Delhi: Dreamtech Press; 2018